

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**

PAT-NO: JP408044661A

DOCUMENT-IDENTIFIER: JP 08044661 A

TITLE: INFORMATION PROCESSOR

PUBN-DATE: February 16, 1996

INVENTOR-INFORMATION:

NAME

IHARA, FUJIO

ASSIGNEE-INFORMATION:

NAME

FUJI XEROX CO LTD

COUNTRY

N/A

APPL-NO: JP06193578

APPL-DATE: July 26, 1994

INT-CL (IPC): G06F013/362, G06F013/362

ABSTRACT:

PURPOSE: To prevent the use efficiency of a bus from being lowered even when there is the bus slave of low-speed processing in a system.

CONSTITUTION: Plural bus masters M<SB>1</SB>-M<SB>3</SB> to request processing, bus slaves S<SB>1</SB>-S<SB>3</SB> for executing requested processing and bus arbitrating means 2 for arbitrating the use right of buses are commonly connected to a shared bus 1. When requesting processing to the bus slaves S<SB>1</SB>-S<SB>3</SB>, the bus masters M<SB>1</SB>-M<SB>3</SB> acquire the bus use rights and issue the processing request commands to the bus slaves and afterwards, those buses are released. On the other hand, the bus slaves temporarily store the received commands addressed to them in queues, select any one of those commands and execute prescribed processing. After the end of processing, an answer command to the bus master of the processing request source is prepared, and a bus using request is issued to the bus arbitrating means 2. The bus arbitrating means 2 preferentially applies the bus use right to the bus slaves S<SB>1</SB>-S<SB>3</SB> rather than the bus masters M<SB>1</SB>-M<SB>3</SB>.

COPYRIGHT: (C)1996,JPO

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平8-44661

(43) 公開日 平成8年(1996)2月16日

(51) Int.Cl. <sup>8</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 13/362	5 1 0 D	9072-5E		
	5 2 0 B	9072-5E		

審査請求 未請求 請求項の数 1 F D (全 6 頁)

(21) 出願番号 特願平6-193578

(22) 出願日 平成6年(1994)7月26日

(71) 出願人 000005496

富士ゼロックス株式会社

東京都港区赤坂三丁目3番5号

(72) 発明者 井原 富士夫

埼玉県岩槻市府内3丁目7番1号 富士ゼ

ロックス株式会社内

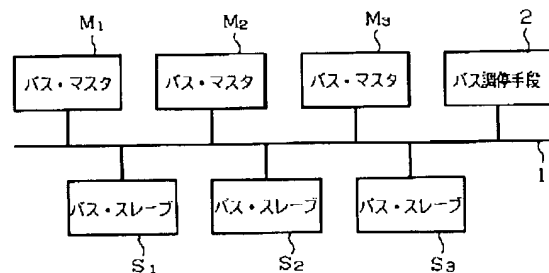
(74) 代理人 弁理士 本庄 富雄

(54) 【発明の名称】 情報処理装置

(57) 【要約】

【目的】 システム内に処理の遅いバス・スレーブがあっても、バスの使用効率が低下しないようにすること。

【構成】 処理を要求する複数個のバス・マスタ  $M_1 \sim M_3$  と、要求された処理を実行するバス・スレーブ  $S_1 \sim S_3$  と、バスの使用権を調停するバス調停手段2とを、共有バス1に共通に接続する。バス・マスタ  $M_1 \sim M_3$  は、バス・スレーブ  $S_1 \sim S_3$  に処理を要求する際には、バス使用権を獲得して処理要求コマンドをバス・スレーブに対して発行した後、バスを解放する。一方、バス・スレーブは、受信した自己宛のコマンドを一旦キューに格納し、その中から1つを選択して所定の処理を実行する。処理終了後は、処理要求元のバス・マスタへの返信コマンドを作成し、バス調停手段2に対してバス使用要求を発行する。バス調停手段2は、バス・マスタ  $M_1 \sim M_3$  よりバス・スレーブ  $S_1 \sim S_3$  に優先的にバス使用権を与える。



## 【特許請求の範囲】

【請求項1】 処理を要求する複数個のバス・マスタと、バス・マスタに要求された処理を実行するバス・スレーブと、それらバス・マスタ、バス・スレーブを共通に接続する共有バスと、該共有バスの使用権を調停するバス調停手段とを有する情報処理装置において、前記バス・マスタは、バス・スレーブに処理を要求する際に、共有バスの使用要求を前記バス調停手段に対して発行するバス使用要求手段と、共有バスの使用権を獲得後、処理を要求するコマンドをバス・スレーブに対して発行し、共有バスを解放するコマンド発行手段とを具え、前記バス・スレーブは、受信したコマンドを溜めておくキューと、共有バスから受信した自己宛のコマンドを前記キューに入れ、また、該キューに溜められたコマンドの中から1つを選択して処理実行部に送るキュー管理手段と、処理実行部の処理終了後に、処理要求元のバス・マスタへの返信コマンドを作成するコマンド発行手段と、処理要求元のバス・マスタに返信コマンドを送るため、前記バス調停手段に対して共有バスの使用要求を発行するバス使用要求手段とを具え、前記バス調停手段は、バス・マスタより優先してバス・スレーブにバス使用権を与えることを特徴とする情報処理装置。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】本発明は、共有バスを有効利用するようにした情報処理装置に関するものである。

## 【0002】

【従来の技術】従来、共有バスに複数のバス・マスタ及びバス・スレーブが接続された情報処理装置において、各バス・マスタにバス使用の優先順位を設定して、各バス・マスタが共有バスを介してデータ転送する際には、バス調停手段により、上記優先順位に基づいてバス使用の調停を行うようにしたものが普及している。

【0003】図6は、バス・マスタ及びバス・スレーブが共有バスに接続された状態を示す図である。図6において、1は共有バス、2はバス調停手段、M<sub>1</sub>～M<sub>3</sub>は、プロセッサ、DMA(Direct Memory Access)コントローラ等のバス・マスタ、S<sub>1</sub>～S<sub>3</sub>は、メモリ、I/O(入出力)装置等のバス・スレーブである。バス・マスタM<sub>1</sub>～M<sub>3</sub>は、バス・スレーブS<sub>1</sub>～S<sub>3</sub>との間でデータ転送を行うに当たり、プログラムからの要求によりバス使用要求を出し、バス使用許可が出たら、共有バス1を使用する。バス調停手段2は、各バス・マスタM<sub>1</sub>～M<sub>3</sub>に設定されたバス使用優先順位に基づいて、各バス・マスタM<sub>1</sub>～M<sub>3</sub>間のバス使用要求の調停を行う。

【0004】例えば、各バス・マスタM<sub>1</sub>～M<sub>3</sub>のバス使用優先順位が、バス・マスタM<sub>3</sub>、M<sub>2</sub>、M<sub>1</sub>の順に高いとすると、バス・マスタM<sub>1</sub>とバス・マスタM<sub>3</sub>がバス調停手段2に対して同時にバス使用要求を出したと

き、バス調停手段2は、バス・マスタM<sub>3</sub>にバス使用許可を与える。しかし、既にバス・マスタM<sub>1</sub>がバス使用中であれば、バス・マスタM<sub>3</sub>がバス調停手段2に対してバス使用要求を出しても、バス・マスタM<sub>1</sub>が共有バス1を解放するまでは、バス・マスタM<sub>3</sub>にバス使用許可は与えられない。しかも、バス・マスタM<sub>1</sub>が共有バス1を解放するのは、バス・スレーブから返信があった後で、例えば、データをバス・スレーブから読み出す場合、バス・スレーブに対して読出コマンドを送った後、バス・スレーブからデータの返送があった時である。

【0005】そのため、バス・マスタM<sub>1</sub>によるデータ読み出しの要求があつてから、データを読み出すのに比較的長い時間を要するバス・スレーブの場合には、その間、他のバス・マスタM<sub>2</sub>、M<sub>3</sub>は共有バス1を使うことができず、共有バス1の使用効率を落としてしまうことになる。特に、高速のマイクロプロセッサの登場により、マイクロプロセッサとメモリとの間でさえその速度差が問題となる最近の状況では、メモリにおいてもそのような問題が発生する。

【0006】そこで、特開平3-51943号公報に示されるように、処理に比較的長い時間を要するバス・スレーブには、コマンドの送信とそれに対する返信とを分割して共有バス1を使用する、分割アクセスを行うようにする技術が提案されている。すなわち、バス・マスタが分割アクセス対応のバス・スレーブにアクセスする時は、該バス・スレーブからの返信を待たずに共有バス1を解放し、該バス・スレーブは、応答できる状態になってからバスの使用権を獲得し、送信元のバス・マスタに対して返信する。そのようにすれば、応答の遅いバス・スレーブからの返信を待っている間に、他のバス・マスタが他のバス・スレーブと通信を行うことができる。

## 【0007】

【発明が解決しようとする課題】しかしながら、前記した従来の技術では、1つのバス・マスタが分割アクセスを実行している時、他のバス・マスタは、他のバス・スレーブにはアクセスできても、分割アクセスされているバス・スレーブにはアクセスできない。そのため、システム内に遅いI/O装置等が複数ある場合には、待たされるバス・マスタの数が多くなってバスの使用効率が低下するという問題点があつた。特に、マイクロプロセッサを複数用いた所謂マルチプロセッサ環境では、その問題がより一層大きくなる。本発明は、以上のような問題点を解決することを課題とするものである。

## 【0008】

【課題を解決するための手段】前記課題を解決するため、本発明では、処理を要求する複数個のバス・マスタと、バス・マスタに要求された処理を実行するバス・スレーブと、それらバス・マスタ、バス・スレーブを共通に接続する共有バスと、該共有バスの使用権を調停するバス調停手段とを有する情報処理装置において、前記バ

ス・マスタは、バス・スレーブに処理を要求する際に、共有バスの使用要求を前記バス調停手段に対して発行するバス使用要求手段と、共有バスの使用権を獲得後、処理を要求するコマンドをバス・スレーブに対して発行し、共有バスを解放するコマンド発行手段とを具え、前記バス・スレーブは、受信したコマンドを溜めておくキューと、共有バスから受信した自己宛のコマンドを前記キューに入れ、また、該キューに溜められたコマンドの中から1つを選択して処理実行部へ送るキュー管理手段と、処理実行部の処理終了後に、処理要求元のバス・マスタへの返信コマンドを作成するコマンド発行手段と、処理要求元のバス・マスタに返信コマンドを送るため、前記バス調停手段に対して共有バスの使用要求を発行するバス使用要求手段とを具え、前記バス調停手段は、バス・マスタより優先してバス・スレーブにバス使用権を与えることとした。

【0009】

【作 用】バス・スレーブでは、共有バスから自己宛のコマンドを受信したら、キュー管理手段により、それをキューに格納する。また、キュー管理手段は、該キューに蓄積されたコマンドの中から1つを選択して処理実行部へ送る。処理実行部の処理が終了したら、コマンド発行手段は、処理要求元のバス・マスタへの返信コマンドを作成する。また、バス使用要求手段は、処理要求元のバス・マスタに返信コマンドを送るため、バス調停手段に対して共有バスの使用要求を発行する。その際、バス調停手段は、バス・マスタより優先してバス・スレーブにバス使用権を与える。

【0010】そのように、バス・スレーブにコマンドを溜めておくキューを設けたため、あるバス・マスタからの要求により処理を行っているバス・スレーブでも、他のバス・マスタから処理要求を受け付けることができる。すなわち、あるバス・マスタが分割アクセスを実行している時、他のバス・マスタは、分割アクセスされているバス・スレーブにもアクセスすることができ、その結果、バス・マスタは、バス・スレーブの処理が終わるまで待たされるということがなくなり、バスの使用効率が低下するということはなくなる。また、バス・マスタからの処理要求コマンドの送信より、バス・スレーブからの返信が優先的に行われるため、その分キューに溜まるコマンド数が減少し、キューの段数を少なくすることができる。

【0011】

【実施例】以下、本発明の実施例を図面に基いて詳細に説明する。図1は、バス・マスタとバス・スレーブの概略の構成を示す図であり、その内、図1(イ)が、バス・マスタの概略の構成を示し、図1(ロ)が、バス・スレーブの概略の構成を示している。符号1、M<sub>1</sub>、S<sub>1</sub>は、図6のものに対応し、3は命令実行手段、4、10はバス使用要求手段、5、11はコマンド発行手段、

6、7はバス監視手段、8はキュー、9はキュー管理手段、12はメモリである。

【0012】バス・マスタM<sub>1</sub>においては、命令実行手段3は、プログラムを解釈・実行し、バス・スレーブS<sub>1</sub>～S<sub>3</sub>にアクセスする必要がある時には、バス使用要求手段4に共有バス1の使用権獲得を依頼し、コマンド発行手段5にコマンドの生成を依頼する。バス使用要求手段4は、命令実行手段3から共有バス1の使用権獲得を依頼されたとき、バス使用要求信号を発行し、バス調停手段2(図6参照)からバス使用許可が得られた時には、それをコマンド発行手段5に通知する。コマンド発行手段5は、命令実行手段3からのコマンド生成依頼を受けてコマンドを生成する。また、バス使用要求手段4からコマンド発行依頼を受けた時に、生成したコマンドを共有バス1上に発行する。バス監視手段6は、常に共有バス1を監視していて、共有バス1を流れるコマンドが自己宛であると判断した時には、それを取得して、それを命令実行手段3に送る。

【0013】また、バス・スレーブS<sub>1</sub>においては、バス監視手段7は、常に共有バス1を監視していて、共有バス1を流れるコマンドが自己宛であると判断した時には、それを取得して、それをキュー管理手段9に伝える。キュー管理手段9は、バス監視手段7が取得したコマンドをキュー8に入れたり、リード/ライト要求コマンドをキュー8から取り出して、処理実行部としてのメモリ12に発行したりする。また、メモリ12からの処理終了を受けてバス使用要求手段10に処理を依頼したり、コマンド発行手段11にメッセージを伝えて返信コマンドの生成を依頼したりする。

【0014】バス使用要求手段10は、キュー管理手段9から共有バス1の使用権獲得依頼を受けた時には、バス使用要求信号を発行し、バス調停手段2(図6参照)から使用許可が得られた時には、それをコマンド発行手段11に伝える。コマンド発行手段11は、キュー管理手段9からのコマンド生成依頼を受けてコマンドを生成する。また、バス使用要求手段10からコマンド発行依頼を受けた時に、生成したコマンドを共有バス1上に発行する。メモリ12は、キュー管理手段9からリード/ライト要求を受けた時にはそれを実行し、要求された処理が完了した時には、データと共にそのことをキュー管理手段9に伝える。

【0015】図2は、コマンド・フォーマットを示す図である。図2に示すように、コマンドは、自己のID(識別子)を書き込むフィールド、相手のIDを書き込むフィールド、リード/ライトの別を書き込むフィールド、相手のアドレスを書き込むフィールド及びメッセージを書き込むフィールドを有している。バス・マスタから発行されるコマンドのメッセージ・フィールドには、ライトの時は、ライトすべきデータが書き込まれ、リードの時には、何も書き込まれない。また、バス・スレー

5

ブから発行される返信コマンドのメッセージ・フィールドには、リードの時には、メモリ12から読み出したデータまたはエラー・メッセージを書き込む。また、ライトの時は、エラーの時以外はコマンドを発行しない。

【0016】図3は、キューの内容の一例を示す図である。キュー8には、受信したコマンドがそのままの形で入っており、図3に示すもののように、例えば、8段の深さを持っているキューであれば、キューの第1段①は次にメモリ12に送られるコマンドが入っているところであり、キューの第8段⑧には、最も後にサービスされ

るコマンドが入るところである。さらに、このキュー8には、現在、キューのどこまでコマンドが入っているかを示すカレント・ポインタがある。

【0017】次に、本発明の動作を説明する。命令実行手段3は、応用プログラムからメモリ12へのアクセスが要求された時には、バス使用要求手段4に共有バス1の使用権獲得を依頼し、コマンド発行手段5にコマンドの生成を依頼する。コマンド発行手段5は、命令実行手段3からのコマンド生成依頼を受けてコマンドを生成する。そして、バス使用要求手段4からコマンド発行依頼を受けて、生成したコマンドを共有バス1上に発行することなく、バスを解放する。

【0018】一方、バス・スレーブ側では、バス監視手段7が自己宛のコマンドを受信したら、それをキュー管理手段9に通知する。該通知を受けたキュー管理手段9では、次のような処理を行う。図4は、キュー管理手段のコマンド格納処理を示すフローチャートである。

ステップ1…バス監視手段7が、バス・マスタM<sub>1</sub>～M<sub>3</sub>からコマンドを受信したか否かを判別する。

ステップ2…受信したら、キュー8の第1段①から第8段⑧までコマンドが入っていて、キュー8が満杯になっているか否かを調べる。

ステップ3…満杯でなければ、カレント・ポインタをインクリメントする。

ステップ4…カレント・ポインタが指す位置に受信したコマンドを入れる。

ステップ5…ステップ2で満杯であれば、ビジー・コマンドの発行をコマンド発行手段11に依頼する。その際、コマンド発行手段11では、図2に示すコマンドの相手のIDのフィールドに、共有バス1上の全てのバス・マスタへの送信を意味する“0”を入れ、メッセージ・フィールドにビジーである旨のメッセージを入れて送出する。

【0019】図5は、キュー管理手段のコマンド実行処理を示すフローチャートである。

ステップ1…キュー8にコマンドが何も入っておらず空になっているか否かを調べる。

ステップ2…空でなければ、キューの第1段①に入っているコマンドをメモリ12に実行させる。

6

ステップ3…メモリ12の処理が終了したか否かを判別する。

【0020】ステップ4…終了したら、処理要求元のバス・マスタへの返信コマンドの生成をコマンド発行手段11に依頼する。この返信コマンドは、図2のものと同様なフォーマットを持ち、自己のIDのフィールドに当該バス・スレーブのID、相手のIDのフィールドに処理要求元のバス・マスタのIDを書き込み、リード/ライトの別を書き込むフィールド、相手のアドレスを書き込むフィールドは空欄とする。また、メッセージ・フィールドには、要求された処理がリードであった時には、メモリ12から読み出したデータまたはエラー・メッセージを書き込む。また、ライトであった時は、書込完了メッセージまたはエラー・メッセージを書き込む。

【0021】ステップ5…バス使用要求手段10に対して、共有バス1の使用要求を行うように処理を依頼する。

ステップ6…カレント・ポインタの値を-1する。

ステップ7…キュー8内にある全てのコマンドをキューの第1段①の方向へ移動させる。

ステップ8…ビジー解除コマンドの発行をコマンド発行手段11に依頼する。その際、コマンド発行手段11では、図2に示すコマンドの相手のIDのフィールドに、共有バス1上の全てのバス・マスタへの送信を意味する“0”を入れ、メッセージ・フィールドにビジー解除のメッセージを入れて送出する。なお、この実施例では、処理を簡略化するため、ビジー・コマンドが発行された後であるか否かに関係なく、1つのコマンドの処理が実行される毎にビジー解除コマンドを発行するようにしたが、ビジー・コマンドが発行された後である場合だけビジー解除コマンドを発行するようにしてもよい。

【0022】バス使用要求手段10は、キュー管理手段9から共有バス1の使用要求を行うように依頼を受けた時には、バス使用要求信号を発行し、その結果、バス調停手段から使用許可が得られたら、それをコマンド発行手段11に伝える。コマンド発行手段11は、キュー管理手段9からのコマンド生成依頼を受けて返信コマンドを生成し、バス使用要求手段10から上記通知を受けた時に、生成した返信コマンドを共有バス1上に発行する。

【0023】なお、上記実施例では、バス・スレーブS<sub>1</sub>の処理実行部がメモリ12である場合で示したが、本発明は、それに限定されずI/O装置等の他のバス・スレーブでも同様に適用可能である。

【0024】

【発明の効果】以上述べた如く、本発明の情報処理装置によれば、バス・スレーブにコマンドを溜めておくキューを設けたため、あるバス・マスタからの要求により処理を行っているバス・スレーブでも、他のバス・マスタから処理要求を受け付けることができるようになった。

7

その結果、バス・マスタは、バス・スレーブの処理が終わるまで待たされるということがなくなり、バスの使用効率が低下するということとはなくなった。また、バス・マスタからの処理要求コマンドの送信より、バス・スレーブからの返信を優先的にを行うようにしたため、その分キューに溜まるコマンド数が減少し、キューの段数を少なくすることができるようになった。

【図面の簡単な説明】

【図1】 バス・マスタとバス・スレーブの概略の構成を示す図

【図2】 コマンド・フォーマットを示す図

【図3】 キューの内容の一例を示す図

8

【図4】 キュー管理手段のコマンド格納処理を示すフローチャート

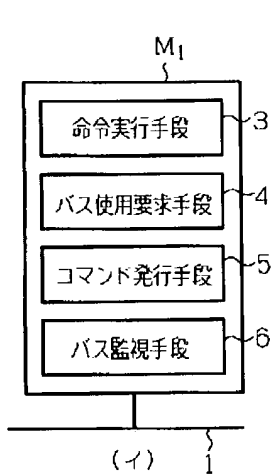
【図5】 キュー管理手段のコマンド実行処理を示すフローチャート

【図6】 バス・マスタ及びバス・スレーブが共有バスに接続された状態を示す図

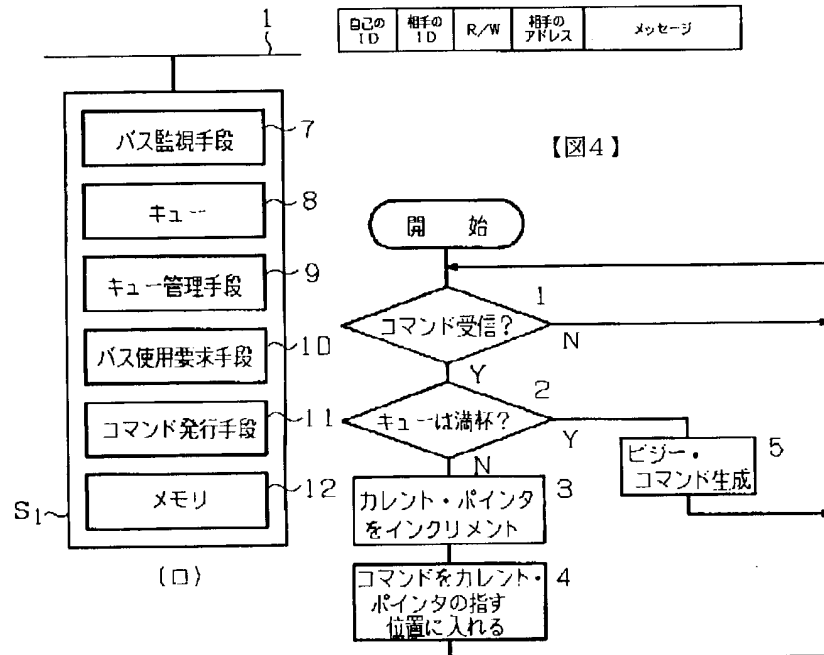
【符号の説明】

1…共有バス、2…バス調停手段、3…命令実行手段、4、10…バス使用要求手段、5、11…コマンド発行手段、6、7…バス監視手段、8…キュー、9…キュー管理手段、12…メモリ、M<sub>1</sub>～M<sub>3</sub>…バス・マスタ、S<sub>1</sub>～S<sub>3</sub>…バス・スレーブ

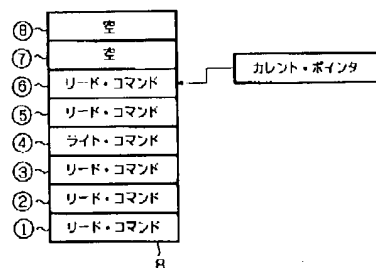
【図1】



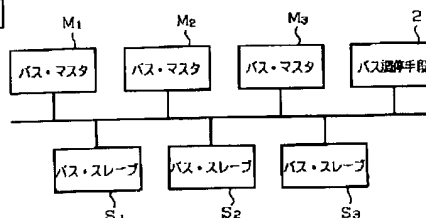
【図2】



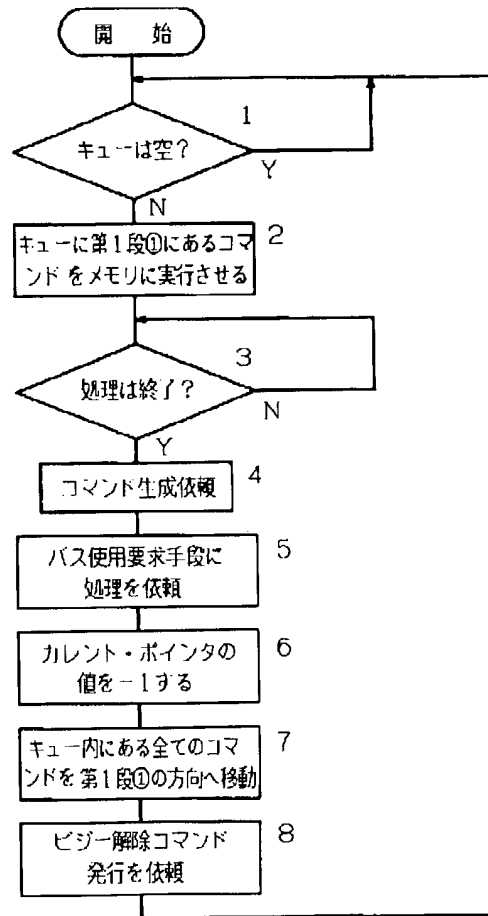
【図3】



【図6】



【図5】





\* NOTICES \*

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

---

CLAIMS

---

[Claim(s)]

[Claim 1] Two or more bus masters which require processing, and the bus slave which performs processing required of the bus master, In the information processor which has the share bus which connects these bus masters and a bus slave in common, and a bus mediation means to arbitrate the royalty of this share bus said bus master A bus activity demand means to publish the activity demand of a share bus to said bus mediation means in case processing is required of a bus slave, The command which requires processing is published to a bus slave after acquiring the royalty of a share bus, and it has a command issuance means to release a share bus. Said bus slave The queue management tool which chooses one from the commands which put the queue which accumulates the received command, and the self-addressed command received from the share bus into said queue, and were accumulated in this queue, and is sent to the processing activation section, In order to send a reply command to a command issuance means to create the reply command to the bus master of processing demand origin after processing termination of the processing activation section, and the bus master of processing demand origin, It is the information processor which is equipped with a bus activity demand means to publish the activity demand of a share bus to said bus mediation means, and is characterized by giving priority to said bus mediation means over a bus master, and granting a bus royalty to a bus slave.

---

[Translation done.]

## \* NOTICES \*

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

---

DETAILED DESCRIPTION

---

[Detailed Description of the Invention]

[0001]

[Industrial Application] This invention relates to the information processor which used the share bus effectively.

[0002]

[Description of the Prior Art] In case the priority of a bus activity is set as each bus master and each bus master carries out data transfer through a share bus conventionally in the information processor by which two or more bus masters and bus slaves were connected to the share bus, what was made to arbitrate the bus activity based on the above-mentioned priority has spread with the bus mediation means.

[0003] Drawing 6 is drawing showing the condition that the bus master and the bus slave were connected to the share bus. Setting to drawing 6, 1 is a share bus and 2 is a bus mediation means and M1 -M3. Bus masters, such as a processor and a DMA (Direct Memory Access) controller, and S1-S3 They are bus slaves, such as memory and I/O (I/O) equipment. Bus master M1 -M3 Bus slave S1 -S3 If in performing data transfer in between a bus activity demand is advanced by the demand from a program and bus licence comes out, the share bus 1 will be used. the bus mediation means 2 -- each -- bus master M1 -M3 the set-up bus activity priority -- being based -- each -- bus master M1 -M3 The bus activity demand of a between is arbitrated.

[0004] for example, -- each -- bus master M1 -M3 bus activity priority -- bus masters M3 and M2 and M1 supposing it is high in order -- bus master M1 Bus master M3 the time of advancing a bus activity demand simultaneously to the bus mediation means 2 -- the bus mediation means 2 -- bus master M3 Bus licence is given. However, it is already a bus master M1. If it is [ bus ] under activity, it is a bus master M3. Even if it advances a bus activity demand to the bus mediation means 2, it is a bus master M1. It is a bus master M3 until it releases the share bus 1. Bus licence is not given. And bus master M1 After there was a reply from a bus slave, for example, after sending a read-out command to a bus slave when reading data from a bus slave, the time of there being return of data from a bus slave releases the share bus 1.

[0005] Therefore, bus master M1 The bus master M2 of others [ case / of the bus slave which takes comparatively long time amount to read data after there is a demand of data read-out to depend ] in the meantime, and M3 The share bus 1 cannot be used but the utilization ratio of the share bus 1 will be dropped. Especially, in the latest situation that the speed difference poses a problem, such a problem generates between a microprocessor and memory also in memory by the appearance of a high-speed microprocessor.

[0006] Then, as shown in JP,3-51943,A, the technique of being made to perform division access which divides transmission of a command and the reply to it into the bus slave which processing takes comparatively long time amount, and uses the share bus 1 for it is proposed. That is, when a bus master accesses the bus slave corresponding to division access, the share bus 1 is released without waiting for the reply from this bus slave, and this bus slave acquires the royalty of a bus, after being in the condition

that it can answer, and answers a letter to the bus master of a transmitting agency. If it is made such, while waiting for the reply from the late bus slave of a response, other bus masters can communicate with other bus slaves.

[0007]

[Problem(s) to be Solved by the Invention] However, in the above mentioned Prior art, while one bus master is performing division access, even if other bus masters can access other bus slaves, they cannot access the bus slave by which division access is carried out. Therefore, when two or more late I/O devices etc. were in a system, there was a trouble that the number of the bus masters kept waiting increased, and the utilization ratio of a bus fell. Especially, in the so-called multiprocessor environment using the microprocessor two or more, the problem becomes still larger. This invention makes it a technical problem to solve the above troubles.

[0008]

[Means for Solving the Problem] Two or more bus masters which require processing in this invention in order to solve said technical problem, In the information processor which has the share bus which connects in common the bus slave which performs processing required of the bus master, and these bus masters and a bus slave, and a bus mediation means to arbitrate the royalty of this share bus A bus activity demand means to publish the activity demand of a share bus to said bus mediation means in case said bus master requires processing of a bus slave, The command which requires processing is published to a bus slave after acquiring the royalty of a share bus, and it has a command issuance means to release a share bus. Said bus slave The queue management tool which chooses one from the commands which put the queue which accumulates the received command, and the self-addressed command received from the share bus into said queue, and were accumulated in this queue, and is sent to the processing activation section, In order to send a reply command to a command issuance means to create the reply command to the bus master of processing demand origin after processing termination of the processing activation section, and the bus master of processing demand origin, It had a bus activity demand means to publish the activity demand of a share bus to said bus mediation means, and it was presupposed to said bus mediation means that priority is given over a bus master and a bus royalty is granted to a bus slave.

[0009]

[work --] for In a bus slave, if a self-addressed command is received from a share bus, it is stored in a queue with a queue management tool. Moreover, a queue management tool chooses one from the commands accumulated in this queue, and sends it to the processing activation section. If processing of the processing activation section is completed, a command issuance means will create the reply command to the bus master of processing demand origin. Moreover, a bus activity demand means publishes the activity demand of a share bus to a bus mediation means in order to send a reply command to the bus master of processing demand origin. In that case, priority is given to a bus mediation means over a bus master, and it grants a bus royalty to a bus slave.

[0010] Since the queue which accumulates the command in the bus slave was formed such, a processing demand is receivable from other bus masters with the bus slave which is processing by the demand from a certain bus master. That is, while a certain bus master is performing division access, if other bus masters also access the bus slave by which division access is carried out, they can do \*\*. Consequently, being kept waiting of a bus master until processing of a bus slave finishes is lost, and the utilization ratio of a bus falls [ stop / \*\*\*\*\* ]. Moreover, from transmission of the processing demand command from a bus master, since the reply from a bus slave is performed preferentially, the number of commands with which the part queue is covered can decrease, and the number of stages of a queue can be lessened.

[0011]

[Example] Hereafter, the example of this invention is explained to a detail based on a drawing. Drawing 1 is drawing showing the configuration of the outline of a bus master and a bus slave, among those drawing 1 (b) shows the configuration of the outline of a bus master, and drawing 1 (b) shows the configuration of the outline of a bus slave. a sign 1, M1, and S1 the thing of drawing 6 -- corresponding

-- 3 -- for a command issuance means, and 6 and 7, as for a queue and 9, a bus monitor means and 8 are [ a bus activity demand means, and 5 and 11 / an instruction-execution means, and 4 and 10 / a queue management tool and 12 ] memory.

[0012] Bus master M1 It sets, the instruction-execution means 3 interprets and executes a program, and it is bus slave S1 -S3. When it is necessary to access, royalty acquisition of the share bus 1 is requested from the bus activity demand means 4, and generation of a command is requested from the command issuance means 5. The bus activity demand means 4 notifies it to the command issuance means 5, when royalty acquisition of the share bus 1 is requested from the instruction-execution means 3, a bus activity demand signal is published and bus licence is obtained from the bus mediation means 2 (refer to drawing 6 ). The command issuance means 5 receives the command generation request from the instruction-execution means 3, and generates a command. Moreover, the command which generated the command issuance request from the bus activity demand means 4 at the time of a carrier beam is published on the share bus 1. The bus monitor means 6 is always supervising the share bus 1, when the share bus 1 is judged that the flowing command is self-addressed, acquires it and sends it to the instruction-execution means 3.

[0013] Moreover, bus slave S1 It sets, and the bus monitor means 7 is always supervising the share bus 1, when the share bus 1 is judged that the flowing command is self-addressed, it acquires it and tells it to the queue management tool 9. The queue management tool 9 puts the command which the bus monitor means 7 acquired into a queue 8, or takes out a read/write demand command from a queue 8, and publishes it in the memory 12 as the processing activation section. Moreover, in response to the processing termination from memory 12, processing is requested from the bus activity demand means 10, or a message is told to the command issuance means 11 and generation of a reply command is requested.

[0014] The bus activity demand means 10 tells it to the command issuance means 11, when a bus activity demand signal is published at the time of a carrier beam and licence is obtained from the bus mediation means 2 (refer to drawing 6 ) in a royalty acquisition request of the queue management tool 9 to the share bus 1. The command issuance means 11 receives the command generation request from the queue management tool 9, and generates a command. Moreover, the command which generated the command issuance request from the bus activity demand means 10 at the time of a carrier beam is published on the share bus 1. Memory 12 tells that with data to the queue management tool 9, when it is performed at the time of a carrier beam and the demanded processing completes a read/write demand from the queue management tool 9 at it.

[0015] Drawing 2 is drawing showing a command format. As shown in drawing 2 , the command has the field which writes in the field which writes in self ID (identifier), the field which writes in a partner's ID, the field which writes in the exception of read/write, the field which writes in a partner's address, and a message. At no times of a light, the data which should be carried out a light are written in, and when it is a lead, it is written in the message field of the command published from a bus master. Moreover, at the time of a lead, the data or the error message read from memory 12 is written in the message field of the reply command published from a bus slave. Moreover, a command is not published other than the time of an error at the time of a light.

[0016] Drawing 3 is drawing showing an example of the content of the queue. If it is the queue which has eight steps of depths in the queue 8 like [ although it gets down from close in the form where the received command is as it is and is shown in drawing 3 ], the command which the command of 1st step \*\* of a queue sent to memory 12 next is just going to be in close, and is served for the 8th step [ of a queue ] \*\* by the very back will just be going to enter. Furthermore, there is a current pointer in which it is shown whether a command is in close by the queue throat top now in this queue 8.

[0017] Next, actuation of this invention is explained. When access to memory 12 from an application program is required, the instruction-execution means 3 requests royalty acquisition of the share bus 1 from the bus activity demand means 4, and requests generation of a command from the command issuance means 5. The command issuance means 5 receives the command generation request from the instruction-execution means 3, and generates a command. And a command issuance request is received

from the bus activity demand means 4, and the generated command is published on the share bus 1. After command issuance releases a bus, without waiting for the reply from a bus master.

[0018] On the other hand, in a bus slave side, if the bus monitor means 7 receives a self-addressed command, it will be notified to the queue management tool 9. With the carrier beam queue management tool 9, the following processings are performed for this advice. Drawing 4 is a flow chart which shows command storing processing of a queue management tool.

the step 1 -- bus monitor means 7 -- bus master M1 -M3 from -- it distinguishes whether the command was received or not.

Step 2 -- If it receives, it will investigate whether a command is in close from the 1st step [ of a queue 8 ] \*\* to the 8th step \*\*, and the queue 8 has filled.

Step 3 -- A current pointer will be incremented if not full.

Step 4 -- The command received in the location which a current pointer points out is put in.

Step 5 -- If full of step 2, issuance of a busy command will be requested from the command issuance means 11. In that case, with the command issuance means 11, "0" which means transmission to all the bus masters on the share bus 1 is put into the field of ID of the partner of the command shown in drawing 2 , and the message of a busy purport is put in and sent out to it in the message field.

[0019] Drawing 5 is a flow chart which shows command execution processing of a queue management tool.

Step 1 -- It investigates whether a command breaks close [ no ] but has become empty at the queue 8.

Step 2 -- If it is not empty, memory 12 will be made to execute the command with which close is in the 1st step \*\* of a queue.

Step 3 -- It distinguishes whether processing of memory 12 was completed.

[0020] Step 4 -- If it ends, generation of the reply command to the bus master of processing demand origin will be requested from the command issuance means 11. Let the field which this reply command has the same format as the thing of drawing 2 , writes ID of the bus master of processing demand origin in the field of self ID in the field of ID the bus slave's concerned, and a partner's ID, and writes in the exception of read/write, and the field which writes in a partner's address be blanks. Moreover, when the demanded processing is a lead, the data or the error message read from memory 12 is written in the message field. Moreover, when it is a light, a write-in completion message or an error message is written in.

[0021] Step 5 -- Processing is requested to the bus activity demand means 10 to perform the activity demand of the share bus 1.

Step 6 -- The value of a current pointer is carried out -one.

Step 7 -- All the commands in a queue 8 are moved in the direction of the 1st step \*\* of a queue.

Step 8 -- Issuance of a busy discharge command is requested from the command issuance means 11. In that case, with the command issuance means 11, "0" which means transmission to all the bus masters on the share bus 1 is put into the field of ID of the partner of the command shown in drawing 2 , and the message of busy discharge is put in and sent out to it in the message field. In addition, in order to simplify processing, after the busy command was published, whenever processing of one command was performed, regardless of whether it is or not, it was made to publish a busy discharge command in this example, and only when it is after a busy command is published, it may be made to publish a busy discharge command.

[0022] The bus activity demand means 10 will tell it to the command issuance means 11, if a bus activity demand signal is published at the time of a carrier beam, consequently licence is obtained from a bus mediation means in a request so that the activity demand of the share bus 1 may be performed from the queue management tool 9. The command issuance means 11 publishes the reply command which received the command generation request from the queue management tool 9, generated the reply command, and generated the above-mentioned advice from the bus activity demand means 10 at the time of a carrier beam on the share bus 1.

[0023] In addition, at the above-mentioned example, it is a bus slave S1. Although shown by the case where the processing activation section is memory 12, this invention is not limited to it but can be

similarly applied with other bus slaves, such as an I/O device.

[0024]

[Effect of the Invention] Since the queue which accumulates the command in the bus slave was formed according to the information processor of this invention as stated above, the processing demand could be received from other bus masters also with the bus slave which is processing by the demand from a certain bus master. Consequently, being kept waiting of a bus master until processing of a bus slave finishes is lost, and the utilization ratio of a bus falls [ stopped / \*\*\*\*\* ]. Moreover, from transmission of the processing demand command from a bus master, in order to perform the reply from a bus slave preferentially, the number of commands with which the part queue is covered can decrease, and the number of stages of a queue could be lessened.

---

[Translation done.]

\* NOTICES \*

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

---

DESCRIPTION OF DRAWINGS

---

[Brief Description of the Drawings]

[Drawing 1] Drawing showing the configuration of the outline of a bus master and a bus slave

[Drawing 2] Drawing showing a command format

[Drawing 3] Drawing showing an example of the content of the queue

[Drawing 4] The flow chart which shows command storing processing of a queue management tool

[Drawing 5] The flow chart which shows command execution processing of a queue management tool

[Drawing 6] Drawing showing the condition that the bus master and the bus slave were connected to the share bus

[Description of Notations]

1 [ -- 5 A bus activity demand means, 11 / -- 6 A command issuance means, 7 / -- A bus monitor means, 8 / -- A queue, 9 / -- A queue management tool, 12 / -- Memory and M1 -M3 / -- bus master and S1 -S3 - bus slave ] -- A share bus, 2 -- A bus mediation means, 3 -- 4 An instruction-execution means, 10

---

[Translation done.]